An Efficient Approach for Solving Large Scale Flexible Job Shop Scheduling Problem: A Strategic Constraint Programming

Mansour Zarrin

1 Introduction

The flexible job shop scheduling problem is a complex combinatorial optimization problem that arises in various industrial and manufacturing scenarios. It involves scheduling a set of jobs, each comprising a sequence of tasks, on a set of machines. Each task can be performed on one of several machines, each with different processing times. The goal is to schedule all tasks such that the makespan — the total time to complete all jobs — is minimized. This model must adhere to several constraints, including task sequencing within each job, the non-overlapping execution of tasks on each machine, and the capacity constraints of each machine.

Flexible Job Shop Scheduling Problem

Parameters:

- J: Set of jobs, indexed by j.
- T_j: Set of tasks for job j, indexed by t.
- M: Set of machines, indexed by m.
- A_{jt}: Set of alternatives for task t of job j, indexed by a.
- d_{jta}: Duration of task t of job j when performed as alternative a.
- m_{jta}: Machine used for task t of job j when performed as alternative a.
- C_m: Capacity of machine m.
- D_{jta}: Demand of task t of job j when performed as alternative a on machine m_{jta}.

Decision Variables:

- $s_{jt} \in R_{\geq 0}$: Start time of task t of job j.
- $e_{jt} \in R_{\geq 0}$: End time of task t of job j.
- $p_{jta} \in \{0, 1\}$: Binary variable indicating if alternative a is chosen for task t of job j.
- $makespan \in R_{\geq 0}$: Total time to complete all jobs.

Objective:

$$\text{Min } makespan \tag{1}$$

Constraints:

Task Execution Constraints:

For each job j and task t:

$$e_{jt} = s_{jt} + \sum_{a \in A_{jt}} p_{jta} \cdot d_{jta} \quad \forall j \in J, \forall t \in T_j \tag{2}$$

The end time of a task is its start time plus its duration. Only the duration of the chosen alternative contributes to the task's duration. For example, if job 1's task 1 can be done on machine A (alternative 1) in 3 hours or on machine B (alternative 2) in 2 hours, and we choose alternative 1, then if $s_{11} = 2$, we must have $e_{11} = 2 + 3 = 5$.

Alternative Selection Constraints:

For each job j and task t:

$$\sum_{a \in A_{jt}} p_{jta} = 1 \quad \forall j \in J, \forall t \in T_j \tag{3}$$

Exactly one alternative must be chosen for each task. In other words, if job 1's task 1 has two alternatives, only one alternative can be chosen, so either $p_{111} = 1$ and $p_{112} = 0$ or $p_{111} = 0$ and $p_{112} = 1$.

Precedence Constraints:

For each job j and consecutive tasks t and $t + 1$:

$$s_{jt+1} \geq e_{jt} \quad \forall j \in J, \forall t \in T_j \setminus \{\text{last task}\} \tag{4}$$

Each task in a job must start after the previous task in the same job has finished. Therefore, when job 1's task 1 ends at time 5 ($e_{11} = 5$), then task 2 of job 1 must start at or after time 5 ($s_{12} \geq 5$).

No Overlap Constraints:

For each machine m, tasks t of job j and t' of job j' cannot overlap if they use the same machine:

$$\text{NoOverlap}(e_{jt}, e_{j't'}) \quad \text{if } m_{jta} = m_{j't'a'} \text{ and } p_{jta} = p_{j't'a'} = 1 \tag{5}$$

No two tasks can be processed on the same machine at the same time. So, if task 1 of job 1 and task 2 of job 2 are both assigned to machine A and task 1 of job 1 is scheduled from 2 to 5, then task 2 of job 2 cannot be scheduled during this time.

Cumulative Constraints:

For each machine m:

$$\text{Cumulative}\left(\{s_{jt}\}, \{d_{jta}\}, \{D_{jta}\}, C_m\right) \quad \text{if } m_{jta} = m \text{ and } p_{jta} = 1 \tag{6}$$

The sum of the demands of the tasks running concurrently on a machine must not exceed the machine's capacity. For example, if machine A has a capacity of 10 and task 1 of job 1 (demand 4) and task 2 of job 2 (demand 7) are assigned to it, they cannot be processed simultaneously because $4 + 7 > 10$.

Makespan Constraints:

$$makespan \geq e_{jt} \quad \forall j \in J, \forall t \in T_j \tag{7}$$

The makespan must be greater than or equal to the end time of the last task of each job. This means, that if the last task of job 1 ends at time 8 and the last task of job 2 ends at time 10, then the makespan must be at least 10.